(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 643 364 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
05.04.2006 Bulletin 2006/14

(51) Int Cl.:
G06F 9/46 *(2006.01)*

(21) Application number: **05019072.7**

(22) Date of filing: **02.09.2005**

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IS IT LI LT LU LV MC NL PL PT RO SE SI SK TR
Designated Extension States:
AL BA HR MK YU

(30) Priority: **30.09.2004 US 614401**

(71) Applicant: **SAP AG**
69190 Walldorf (DE)

(72) Inventors:
- **Kahn, Markus, Dr.**
  69120 Heidelberg (DE)
- **Baumann, Marcus**
  69207 Sandhausen (DE)

(74) Representative: **Oppermann, Frank**
Luderschmidt, Schüler & Partner
Patentanwälte
John-F-Kennedy-Strasse 4
65189 Wiesbaden (DE)

(54) Systems and methods for general aggregation of characteristics and key figures

(57) The present invention refers to a computer-implemented method, a computer system, and a computer program product for automated generic and parallel aggregation of characteristics and key figures of unsorted mass data being of specific economic interest, particularly associated with financial institutions, and with financial affairs in banking practice, said parallel aggregation reducing the amount of data for a customer defined granularity for the purpose of facilitating the handling of raw data related to all areas of credit risk management in banking practice. Moreover, said method improves the computing power of software and the software performance (run time), respectively, preferably in the case of mass data.

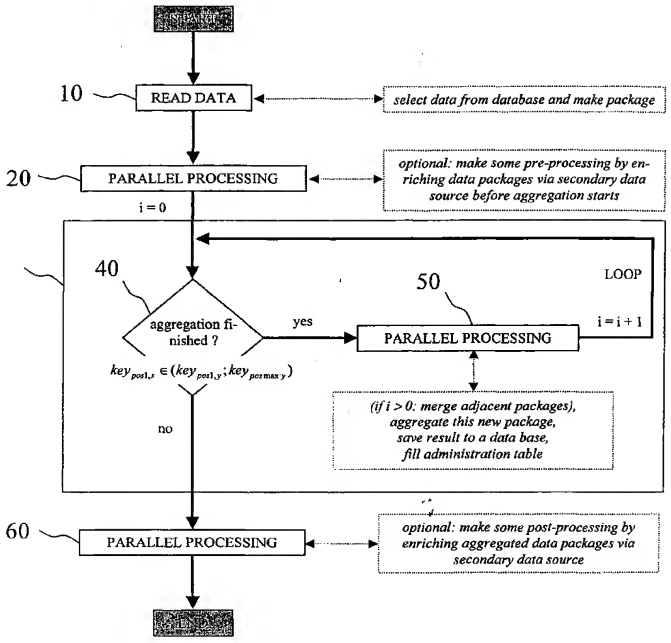

Fig. 2

EP 1 643 364 A1

Description

FIELD OF THE INVENTION

[0001] The present invention generally relates to electronic data processing, and in particular, to a computer-implemented method, computer system and computer program product for automated generic and parallel aggregation of characteristics and key figures of mass data associated with financial institutions and with financial affairs in banking practice.

BACKGROUND OF THE INVENTION

[0002] As international financial markets expand, global concerns over the soundness of banking practices are driving stringent new requirements for bank-level management, regulatory control, and market disclosure.

[0003] Prior art data processing systems in banking are provided with software tools, for example SAP proprietary software tool solutions in banking, e.g. the SAP solution for the new Basel Capital Accord (Basel II) that builds on the proven capabilities of the SAP for Banking solution portfolio, that enable to pursue said requirements.

[0004] The SAP solution for the new Basel Capital Accord (Basel II) represents a risk-sensitive framework that provides capabilities for calculating risk exposure and capital, for managing market risk, interest risk, or liquidity risk, and for calculating and managing all areas of credit risk, helping to facilitate the handling of mass data, particularly being of specific economic interest and associated with financial institutions and with financial affairs in banking practice.

[0005] Moreover, software tool solutions for banking systems including capabilities for computing descriptive statistics are needed to efficiently analyze large amounts of given data (mass data) while managing large and complex projects. Within that scope, mass data are often required to be aggregated according to a customer defined granularity. Accordingly, aggregations can be computed for characteristics (lexicographic min, max) and key figures (min, max, count, sum, avg, variance, std, var%) using prior art software tool solutions.

[0006] In view of prior art software tool solutions for banking systems, there still remains the need to improve the computing power of software and software performance (run time performance), respectively, in particular, when it comes to large amounts of data (mass data) to be aggregated effectively that can not be handled in the main memory of a data processor.

SUMMARY OF THE INVENTION

[0007] The present invention meets the above-identified need by providing an adequate computer-implemented method for automated generic and parallel aggregation of characteristics and key figures of mass data, particularly associated with banking practice, that can be easily integrated into existing credit risk platforms as, for example, the above mentioned SAP solution for Basel II.

[0008] It is another object of the present invention to provide a computer system and a computer program product for automated generic and parallel aggregation of characteristics and key figures of said mass data, and further a data carrier readable by a computer, the data carrier storing a plurality of instructions implemented by a computer program for causing the processing means of a computer system to execute the computer-implemented method.

[0009] Moreover, it is an object of the present invention to provide a computer-implemented method for automated generic and parallel aggregation of characteristics and key figures of mass data associated with banking practice, that are not assumed to be a priori sorted in respect to a free selectable granularity before applying said computer-implemented method.

[0010] A further object of the present invention is to provide a computer-implemented method that can optionally perform the automated generic aggregation of data either in linear or in parallel processing mode, thereby noticeably improving the computing power of software, as preferably in the case of mass data, depending on the capacity utilization of a data processing system.

[0011] To achieve the foregoing objects, and in accordance with the purpose of the invention as embodied and broadly described herein, there is provided a computer-implemented method for automated generic and parallel aggregation of characteristics and key figures of mass data whose structure is unknown, particularly associated with financial institutions and with financial affairs in banking practice, provided by different databases of different data sources, said method reducing the amount of data to a customer defined granularity by computing aggregations on key figures within the scope of an iterative process, repeatedly processing a parallel aggregation algorithm including parallel processing steps for merging, reorganizing, sorting and aggregating data records.

[0012] In another aspect of the invention, the aggregation is computed on predetermined key figures using predetermined aggregation operations selected from a function pool and / or costumer defined aggregation operations to be defined by input means using said predetermined aggregation operations.

[0013] In yet another aspect of the invention, the aggregation is computed on costumer defined key figures to be defined by input means using said predetermined aggregation operations selected from a function pool and using said predetermined aggregation operations and / or said costumer defined aggregation operations.

[0014] In yet another aspect of the invention, the aggregation algorithm can run in parallel processing mode for mass data, thereby noticeably improving the computing power of software, but if required, depending on the capacity utilization of a data processing system, the processing of said aggregation algorithm can optionally run in linear processing mode.

[0015] In yet another aspect, the aggregation algorithm of the present invention can easily be integrated into other processes, e.g. as a pre-processing before a data extraction of business area information to a business information warehouse of a company, thereby separating the results of already aggregated mass data for the purpose of visualizing data of specific economic interest.

[0016] Alternatively, the aggregation algorithm of the present invention can be applied to prior art software solutions in the context of an ad hoc reporting for descriptive statistics.

[0017] These and other features, objects, and advantages of the preferred embodiments will become apparent when the detailed description of the preferred embodiments is read in conjunction with the drawings attached hereto.

BRIEF DESCRIPTION OF DRAWINGS

[0018]

Fig. 1 illustrates a schematic view of the computer-implemented method for automated generic and parallel aggregation of characteristics and key figures of unsorted mass data;

Fig. 2 illustrates a simplified flow chart of the computer-implemented method showing the method steps for automated generic and parallel aggregation of characteristics and key figures of unsorted mass data;

Fig. 3 illustrates the flow chart showing the method steps for the aggregation of records within a single data package;

Fig. 4a illustrates an example of use for raw data, showing a work list of M = 12 data records associated with financial institutions and with financial affairs in banking practice;

Fig. 4b illustrates granularity characteristics / granularity levels i of granularity characteristics;

Fig. 5 illustrates an example of use for the parallel aggregation algorithm illustrated in Fig. 2, wherein the original amount of data records shown in Fig. 4a is reduced to N = 4 < M = 12 data records for a customer defined granularity as it is set out in Fig. 5 referring to "search result"; and

Fig. 6 illustrates an example of use for the parallel aggregation algorithm illustrated in Fig. 2, wherein the original amount of data records shown in Fig. 4a is reduced to N = 4 < M = 12 data records for a customer defined granularity as it is set out in Fig. 5 referring to "search result", and wherein another compromise of performance is made compared to the preceding example of use of Fig. 5.

DETAILED DESCRIPTION OF EXEMPLARY EMBODIMENTS

[0019] Reference will now be made in detail to the present invention, examples of which are illustrated in the accompanying drawings in which like reference numbers refer to corresponding elements.

[0020] The present invention does not only refer to a computer-implemented method for automated generic and parallel aggregation of characteristics and key figures of unsorted mass data associated with financial institutions and with financial affairs, but also to a data processing system, a computer program product that can be stored on a computer readable data carrier, and a data carrier.

[0021] The data processing system (computer system) may comprise a single data processor or a plurality of data processors via inter-computer network, each data processor including processing means (processor), storage means (memory), bus means (bus), network means (network), interface means, input means and output means (input and output devices). The computer system may also be simply a server.

[0022] The data processor is, for example, a conventional desktop Computer, a multiprocessor computer, or the like. The Processor is, for example, a Central Processing Unit (CPU), a Micro Controller Unit (MCU), Digital Signal Processor (DSP), or the like.

[0023] Storage means are in particular provided for storing said specified mass data. Storage means symbolizes any memory means for temporarily or permanently storing data and instructions. Although memory is conveniently illustrated

as part of computer, memory function may also be implemented in network, in computers and in processor itself, e.g. cache, register, or elsewhere. Memory can be, for example, a Read Only Memory (ROM), Random Access Memory (RAM), or a memory with other access options. Memory is physically implemented by computer-readable media, for example: (a) magnetic media, such as hard disk, floppy disk or other magnetic disk, tape or cassette tape; (b) optical media, such as optical disk (CD-ROM, DVD); (c) semiconductor media, like DRAM, SRAM, EPROM, EEPROM, or the like.

[0024] Memory means may further store support modules, for example, a Basic Input Output System (BIOS), an Operating system (OS), a program library, a compiler or interpreter, and a text processing tool.

[0025] Input means symbolizes any device for providing data and instructions for processing by computer, for example, a keyboard or pointing device such as a mouse, trackball or cursor direction key.

[0026] Output means symbolizes any device for presenting results of aggregated data packages, for example, a monitor or a display, for example, a Cathode Ray Tube (CRT), Flat Panel Display, Liquid Crystal Display (LCD), or printer.

[0027] Bus and network provide logical and physical connections by conveying data and instruction signals. While connections inside computer are conveniently referred to as "bus", connections between computers are referred to as "inter-computer network". Optionally, network comprises gateways being devices (computers) that specialize in data transmission and protocol conversion, allowing users working in one network to access another network.

[0028] Networking environments (as network) are commonplace in offices, enterprise-wide computer networks, intranets and the internet (i.e. world wide web). Network can be a wired or wireless network. To name a few network implementations, network is, for example, a local area network (LAN), a wide area network (WAN), a public switched telephone network (PSTN), an Integrated Services Network (ISDN), an infra-red (IR) link, a radio link, like Universal Mobile Tele-communications System (UMTS), Global System for Mobile Communication (GSM), Code Division Multiple Access (CDMA), or satellite link.

[0029] Transmission protocols and data formats are known as, for example, transmission control protocol/internet protocol (TCP/IP), hyper text transfer protocol (HTTP), secure HTTP, wireless application protocol, unique resource locator (URL), unique resource identifier (URI), hyper text markup language HTML, extensible markup language XML, extensible hyper text markup language XHTML, wireless application markup language (WML), etc.

[0030] Interface means (interfaces) for linking together the data processing units of a data processing system are well known in the art. An interface can be, for example, a serial port interface, a parallel port interface, a universal serial bus (USB) interface, an internal or external modem.

[0031] The computer program product comprises a plurality of instructions for causing the processing means of a computer system to execute the method steps of the invention specified hereinafter with more detail. In other words, computer program product defines the operation of computer and its interaction in inter-computer network. For example, computer program product may be available as source code in any programming language, and as object code (binary code) in a compiled form. Persons skilled in the art can use computer program product in connection with any of support modules (e.g. compiler, interpreter, operating system). The computer program product is stored in memory hereinafter referred to as data carrier.

[0032] For the communication of computer program product and computer, data carrier is conveniently inserted into input device. Data carrier is implemented as any computer readable medium. Generally, carrier is an article of manufacture comprising a computer readable medium having readable program code means embodied therein for executing the method steps of the present invention. Furthermore, program signal can also embody computer program. Program signal is transmitted via inter-computer network to data processor.

[0033] Fig. 1 illustrates a schematic view of the computer-implemented method for automated generic and parallel aggregation of characteristics and key figures of unsorted mass data in particular being of specific economic interest and associated with financial institutions and with financial affairs in banking practice. The mass data ("input data") whose structure is unknown include a plurality of M data records, wherein M represents a large amount of data records to be aggregated that can not be handled in the main memory of a data processor. The mass data ("input data") further consist of packetized blocks of data provided by different databases of different accessible data sources, including sets of rows and sets of columns, each row corresponding to a record, and the columns including fields of predetermined granularity characteristics and fields of predetermined key figures. Generally speaking, the generic aggregation of characteristics and key figures aims at the reduction of said mass data according to a given customized granularity. Due to the plurality of M data records, said mass data are customized as packages including $M_p < M$ data records as it is illustrated in the upper block of Fig. 1 referred to as "Built packages" before being assigned to the parallel aggregation algorithm. The built data packages (package 1, package 2, ...package n) are assigned to different jobs so that each job includes a plurality of data packages. A job or a plurality of jobs can be processed in a parallel processing mode, thereby noticeably improving the computing power and run time performance of software, respectively, either using a single data processor or a network of data processors by applying the method steps illustrated in the lower block of Fig. 1. But if required, depending on the capacity utilization of a data processing system, the processing of said aggregation algorithm can optionally run in linear processing mode, thereby aggregating and merging packages within a job sequentially. The method steps of the aggregation algorithm illustrated in the lower block of Fig. 1 are explained in detail below.

[0034] Fig. 2 illustrates a simplified flow chart of the computer-implemented method showing the method steps for automated generic and parallel aggregation of characteristics and key figures of unsorted mass data.

[0035] In method step 10, the computer-implemented method begins with a selection of investigated mass data ("input data") including said plurality of M data records to be aggregated, said mass data being provided by different accessible primary databases of different accessible data sources. Having finished the selection of mass data due to selection criteria, the variously selected blocks of packetized mass data are assigned among each other and the result of assignment is stored to a global database.

[0036] Thereupon, some customizing of the selected mass data is required for defining granularity characteristics and aggregation operations to be carried out by the processing means of a data processing system for computing fields of key figures.

[0037] Moreover, the selected mass data are prepared as data packages according to a customer defined package size including $M_p < M$ data records in a pre-processing step before reading said mass data into the processing means of a data processing system.

[0038] In method step 20, the packaged data can be additionally enriched in a parallel pre-processing step 20 with data from an accessible single secondary database or from accessible secondary databases, subsequently saving the results of enrichment to those local databases of the respective data processors where the data are to be processed.

[0039] Subsequent to the packaging, the data packages are read into the data processing means of a data processing system to be processed within jobs, each of the jobs including a plurality of data packages.

[0040] A job or a plurality of jobs can be processed in a parallel processing mode either using a single data processor or a network of data processors.

[0041] Up to this point, there is not identified one granularity level i corresponding to i = 0. By assigning the data packages to the parallel aggregation algorithm 30, a first parallel processing begins with method step 40, wherein at first said customized granularity characteristics are identified so as to obtain levels i of granularity characteristics. Having identified said granularity characteristics within said data packages, thereby accomplishing the criterion i > 0, the records of each data package are sorted for a given order of said granularity characteristics, and subsequently aggregated for said key figures by using customized aggregation operations, thereby reducing the amount of records in said data packages to $N_a < M_p$ (smaller than the maximum size M_p). Following the aggregation, the results of each aggregated data package are saved to those local databases of the respective data processors where the data are processed. Thereupon, the aggregated packages are split into several smaller sub packages including N_{sp} data records and the size (number of records) and the first and the last record of each sub data package is stored to a global result database. Hereafter, the identification of adjacent packages based on these small sub data packages is executed by checking the termination criterion for the loop i = i + 1 ("not in parallel") being:

if $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate**,

wherein *pos1* illustrates the first position of a data package, *posmax* illustrates the last position of a data package, and *x, y* illustrates the number of a data package, thereby comparing the key of the first record of each data package with the first and the last record of all the rest of data packages (thus comparing all combinations x, y). If said criterion for terminating the loop i = i + 1 is not accomplished, meaning that the conditional inquiry is true, the data packages are assigned for rebuilding new data packages.

[0042] The underlying idea of splitting aggregated data packages into sub data packages is to improve the expressiveness of key information, and thereby to improve the identification of adjacent data packages based on their respective key information. Since only the data package size and the key information of the first and the last record of each data package are stored to a global database while all other data records are not considered, the following interests working in opposite directions must be kept in mind. While large package sizes are ideal for aggregation, the key information of the first and the last record of each large data package is not representative for all the rest of data records within said data package. On the other hand, if the data packages are very small, then the first and last record of each data package is more or less representative for all the rest of data records. But by reducing package sizes, the efficiency of aggregation diminishes due to the fact that there is not much to aggregate in small data packages.

[0043] Thus, the point is to meet the above identified two interests working in opposite direction by approaching an efficient compromise of performance allowing to aggregate relatively large data package sizes, and subsequently split the aggregated data packages into smaller sub packages for the purpose of identifying adjacent sub data packages.

[0044] The relation of the maximum data package size M_p and the size of sub data packages N_{sp} depends on the degree of fragmentation and the degree of aggregation of the unsorted input data.

[0045] The effect of this approach of splitting aggregated data packages into sub data packages becomes the more important the less sorted the input data are, and the lower the degree of aggregation is, or in other words, the lower the reduction of the number of data records is.

[0046] In method step 50, the aggregated packages are assigned to a second parallel processing of the aggregation algorithm 30 for merging adjacent packages, thereby rebuilding new data packages, wherein adjacent packages are those packages with keys of the first record which are closest together. By merging theses small data packages the

maximum allowed package size M_p is restored again. Hereupon, the new data packages (merged packages) are stored to local databases of the respective data processors where the data are processed. Subsequently, the new data packages are assigned again to the above mentioned first parallel processing for reorganizing and sorting, and thereafter aggregating said new data packages for key figures by using said customized aggregation operations.

[0047] After each loop cycle, the conditional inquiry for terminating the loop i = i + 1 is checked for all combinations (x, y) anew, repeatedly executing the loop i = i + 1 while the termination criterion is not accomplished, meaning that the conditional inquiry is true, else, after accomplishing said criterion, i.e. all the data packages are disjoint with regard to the granularity characteristics, terminating the loop.

[0048] Finally, the packaged data can be additionally enriched in a parallel post-processing step 60 with data from an accessible secondary database or from accessible secondary databases, subsequently saving the results of data packages to a global result database.

[0049] Fig. 3 illustrates a flow chart showing the method steps for the aggregation of records within a single data package after entering the aggregation algorithm 30 of Fig. 2. At first, in method step 70, there is not identified one level of granularity characteristics, which is symbolized by i = 0. Furthermore, before aggregating for the first time within the scope of a first iteration, the summary table referred to as itab in which the aggregation result is stored to is empty. At this point, the records within the data package are assigned to a first parallel processing, wherein the fields of granularity characteristics are identified according to a customer defined granularity so as to obtain levels i (i = 1 ....n) of identified granularity characteristics. Having identified said granularity characteristics, thereby accomplishing the criterion i > 0, the records of said data package are sorted for a given order of said granularity characteristics as, for example, illustrated in Fig. 5 referring to "search result". By entering the loop 85, the records are assigned to the approach for sequentially aggregating the unique granularity levels i using predetermined and customized aggregation operations. Beginning with the first granularity level i = 1 in method step 80, the level i = 1 is compared with the maximum level n inquiring the condition being (i > n ?) in method step 90. As long as the condition (i > n ?) is not accomplished, meaning that the conditional inquiry is false, and thus i having a value less than or equal n, in method step 100, the records of the data package tab(i) corresponding to the appropriate granularity level i = 1 are aggregated for a specific key figure Xj by using predetermined aggregation operations (operator j), thereby entering an internal loop 95. Subsequently, in method step 110, the aggregated key figure Xj is moved to the structure str1. Thereafter, in method step 120, it is inquired if the aggregation of data records for key figures in respect to the appropriate granularity level i = 1 is completed. If the conditional inquiry is not accomplished, the records of the data package corresponding to level i = 1 are assigned again to a subsequent aggregation in respect to another key figure using another operation, repeatedly executing this approach of aggregation steps until all selected aggregation operations are conducted, else, leaving the internal loop 95. In method step 130, customer defined aggregation operations can be applied using SAP-BAdI aggregation technique, subsequently saving the results to the structure str1, wherein previous results may be changed. Thereupon, having completed the aggregation algorithm for the appropriate level i = 1, the structure str1 is appended to the summary table itab. This approach for executing the loop 85 is to be applied to all remaining granularity levels i up to and including the maximum number of i (i = 2, 3, 4 ....n). By accomplishing said criterion in method step 90 for leaving the loop 85 being (i > n ?), in method step 150, a global administration table is filled with itab-information. Finally, in method step 160, the summary table referred to as itab is saved to a local database.

[0050] Fig. 4a illustrates an example of use for raw data showing a work list including M = 12 records associated with financial institutions and with financial affairs in banking practice to be applied to the parallel aggregation algorithm 30 of Fig. 2. The work list includes sets of rows and sets of columns, each row corresponding to a record, and the columns including fields of predetermined granularity characteristics, and fields of predetermined key figures.

[0051] Furthermore, the records are sorted according to a given order of granularity characteristics as set out in Fig. 4b under "granularity characteristics / "granularity levels i of granularity characteristics".

[0052] Fig. 5 illustrates an example of use for the aggregation using a processing tool based on the parallel aggregation algorithm 30 of Fig. 2. The aggregation of the raw data illustrated in Fig. 4a including M = 12 data records reduces the amount of data to 4 < M = 12 data records according to the customer defined granularity, as it is set out in Fig. 5 referring to "search result".

[0053] The granularity fields including granularity characteristics are characterized by "rating method" and "rating segment". The fields of key figures are characterized by the columns "financial statement key figure 1" and "financial statement key figure 2".

[0054] The data package size is determined through customizing. Contrary to the preceding statement that large data package sizes are ideal for aggregation, whereas small data package sizes are ideal for reorganization, in this example of use only one single package size can be determined, meaning that the data package size M_p for aggregation is identical to the sub data package size N_{sp} for reorganization. Therefore, in this example of use a less efficient compromise of performance has to be chosen to meet said opposite demands. The customized package size is determined by M_p = 4 corresponding to the maximum number of granularity levels i, as it is shown below in Table 1 and in Fig. 5 referring to "search result", respectively.

[0055] In method step 200, the raw data shown in the original work list of Fig. 4a are exemplarily arranged by the key figures in the column "financial statement key figure 1" in ascending order so as to demonstrate a work list of unsorted records to begin with. Due to the customized data package size of $M_p = 4$, the M = 12 data records of said work list are split in three data packages, in data package 1, data package 2 and data package 3, each data package as a result having 4 data records.

[0056] Furthermore, for the exemplification of the parallel aggregation algorithm as illustrated in Fig. 2 on the basis of the concrete example and to simplify matters, only the granularity fields characterized by "rating method" and "rating segment", the fields of key figures characterized by the columns "financial statement key figure 1" and "financial statement key figure 2", and the field currency are taken into consideration. All the rest of fields remain empty. Hereinafter, Table 2 illustrates the outcome of this reorganization and simplification of said original work list shown in Fig. 4a.

Table 2

Step: 200

rating method	rating-segment	financial statement key figure 1	financial statement key figure 2	currency	package
insurances	life insurances	1620	865860	EUR	1
credit institutions	Landesbanken *(form of banks)*	1912	809485	EUR	1
credit institutions	Sparkassen *(form of banks)*	2860	456825	EUR	1
credit institutions	Sparkassen *(form of banks)*	3254	693677	EUR	1
insurances	casualty insurances	3346	729541	EUR	2
credit institutions	Landesbanken *(form of banks)*	3393	542616	EUR	2
insurances	life insurances	5966	670365	EUR	2
credit institutions	Landesbanken *(form of banks)*	6135	166310	EUR	2
credit institutions	Sparkassen *(form of banks)*	8149	484449	EUR	3
insurances	casualty insurances	8683	824001	EUR	3
insurances	life insurances	8715	247374	EUR	3
insurances	casualty insurances	8916	35040	EUR	3

[0057] In method step 210, the data packages are assigned to the parallel processing of the aggregation algorithm 30 of Fig. 2. Within the scope of a first iteration (Iteration Nr. 1), the parallel processing begins with the method step 40 of Fig. 2. Up to this point, there is not identified one granularity level i mentioned above, which is symbolized by i = 0 in Fig. 2. Therefore, at first, the fields of granularity characteristics labeled "rating method" and "rating segment" are identified so as to obtain levels i of granularity characteristics within said data packages, thereby accomplishing the criterion i > 0. The maximum reachable number of granularity levels i per data package is i = 4 due to Table 1 mentioned above. By sequentially comparing the above mentioned granularity characteristics shown in Table 1 with the data records of each of the three data packages, thereby beginning with the first row of granularity characteristics of Table 1 characterized through **"credit institutions / private banks"** and ending with the forth row characterized through **"insurances / casualty insurances"**, in the example of use there appear in total three matches in each of the three data packages, what as a result corresponds to a granularity level of i = 3 of identified granularity characteristics for each data package.

[0058] Subsequently, the data records within all of the three data packages are sorted according to the given order as set out above in Table 1. The outcome of this sorting is illustrated below in Table 3.

Table 3

Step: 200

rating method	rating-segment	Sum financial statement key figure 1	Minimum financial statement key figure 2	currency	package
credit institutions	private banks	1912	809485	EUR	1
credit institutions	public banks	3254	693677	EUR	1
credit institutions	public banks	2860	456825	EUR	1
insurances	life insurances	1620	865860	EUR	1
credit institutions	private banks	6135	166310	EUR	2
credit institutions	private banks	3393	542616	EUR	2
insurances	life insurances	5966	670365	EUR	2
insurances	casualty insurances	3346	729541	EUR	2
credit institutions	public banks	8149	484449	EUR	3
insurances	life insurances	8715	247374	EUR	3
insurances	casualty insurances	8683	824001	EUR	3
insurances	casualty insurances	8916	35040	EUR	3

[0059] As illustrated in Table 3, the first two rows of data package 1 and data package 2 appear to have identical granularity characteristics. In data package 3, the last two rows include identical granularity characteristics.

[0060] Thereafter, these rows are aggregated for the key figures (Xj) characterized through "financial statement key figure 1" and "financial statement key figure 2" by applying appropriate aggregation operations (operators j) to the respective key figures, wherein said aggregation operations being predetermined or customized aggregation operations. In this case the matches are added up in respect to key figure 1, and in respect to key figure 2 the minimum value is taken over. All three data packages are processed simultaneously due to the parallel processing. As a result, the number of data records within all of the three data packages is reduced to $N = 3 < M_p = 4$ data records, which is illustrated below in Table 4.

Table 4

Step: 210

rating method	rating-segment	Sum financial statement key figure 1	Minimum financial statement key figure 2	currency	package
credit institutions	private banks	1912	809485	EUR	1
credit institutions	public banks	6114	456825	EUR	1
insurances	life insurances	1620	865860	EUR	1
credit institutions	private banks	9528	166310	EUR	2
insurances	life insurances	5966	670365	EUR	2
insurances	casualty insurances	3346	729541	EUR	2
credit institutions	public banks	8149	484449	EUR	3
insurances	life insurances	8715	247374	EUR	3
insurances	casualty insurances	17599	35040	EUR	3

[0061] Thereupon, in step 220, after saving the results of each data package in a database, the identification of adjacent data packages is conducted by checking the termination criterion of the loop i = i + 1 ("not in parallel") being:
if $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate,**
wherein ***pos1*** illustrates the first position of a data package, ***posmax*** illustrates the last position of a data package, and *x, y* illustrates the number of a data package, thereby comparing the key of the first record of each data package with the key of the first and the last record of all the rest of data packages (thus comparing all combinations x, y). If said criterion is not accomplished, meaning that the conditional inquiry is true, the data packages are assigned for rebuilding new data packages.

[0062] Beginning with the key of the first record of data package 1, the comparison of data package 1 and data package 2 results in that the key of the first record of data package 1 is equal to the key of the first record of data package 2. As a result, interpreting the conditional inquiry for the loop, the key of the first record of data package 1 is an element of the amount of data in data package 2, or furthermore interpreted, data package 1 and data package 2 intersect, and thus they are identified as adjacent packages. Consequently, as the termination criterion for the loop is not accomplished, data package 1 and data package 2 are assigned for rebuilding a new data package 1. Since data package 1 and data package 2, respectively, include 3 records, the data package size of the new data package 1 including $M_m = 6$ records exceeds the maximum package size of $M_p = 4$, which is acceptable. The data package 3 remains unmodified.

[0063] In step 230, the aggregated data packages are assigned to the second parallel processing of the aggregation algorithm 30 of Fig. 2 illustrated by the method step 50 within the scope of a second iteration (Iteration Nr. 2) for merging said adjacent data packages of step 210. Having merged said data package 1 and data package 2 to a new data package 1, the data records of the remaining two data packages are assigned again to the above mentioned first parallel process illustrated by method step 40 of Fig. 2 within the scope of a second iteration (Iteration Nr. 2), wherein the data records of the remaining two data packages are reorganized in parallel processing mode, and thereafter sorted again according to the given order for said granularity characteristics as illustrated in Table 1 and in Fig. 4b, respectively. This outcome of this reorganization and sorting is illustrated hereinafter in Table 5.

Table 5

Step: 230

rating method	rating-segment	Summe financial statement key figure 1	Minimum financial statement key figure 2	currency	old package	new package
credit institutions	private banks	1912	809485	EUR	1	1
credit institutions	private banks	9528	166310	EUR	2	1
credit institutions	public banks	6114	456825	EUR	1	1
insurances	life insurances	1620	865860	EUR	1	1
insurances	life insurances	5966	670365	EUR	2	1
insurances	casualty insurances	3346	729541	EUR	2	1
credit institutions	public banks	8149	484449	EUR	3	2
insurances	life insurances	8715	247374	EUR	3	2
insurances	casualty insurances	17599	35040	EUR	3	2

[0064] Thereupon, the aggregation for said key figures using said predetermined aggregation operations is conducted anew, wherein as a result, the size of the new data package 1 decreases from $M_m = 6$ to $N_a = 4$ according to the customer defined granularity as illustrated in Fig. 5 referring to "search result". Following the aggregation, the results of the remaining data packages are saved in a database. The outcome of this aggregation is illustrated hereinafter in Table 6.

Table 6

Step: 230

rating method	rating-segment	financial statement key figure 1	financial statement key figure 2	currency	package
credit institutions	private banks	11400	166310	EUR	1
credit institutions	public banks	6114	456825	EUR	1
insurances	life insurances	7586	670365	EUR	1
insurances	casualty insurances	3346	729541	EUR	1
credit institutions	public banks	8149	484449	EUR	2
insurances	life insurances	8715	247374	EUR	2
insurances	casualty insurances	17599	35040	EUR	2

[0065] In step 240, the termination criterion for the loop i = i + 1 for the remaining two data packages is checked anew ("not in parallel"). In this case, the comparison of data package 1 and data package 2 results in that the key of the first record of data package 2 is greater than the key of the first record of data package 1, and that the key of the last record of data package 1 is greater than said key of the first record of data package 2, which represents intersecting data packages. As a result, the termination criterion is not accomplished, consequently assigning the data package 1 to data package 2 for rebuilding a new data package 1. Since data package 1 includes 4 records and data package 2 includes 3 records, the data package size of the new data package 1 including $M_m = 7$ records exceeds the maximum package size of $M_p = 4$, which is acceptable.

[0066] In step 250, the aggregated data packages are assigned again to the second parallel processing of the aggregation algorithm 30 of Fig. 2 illustrated by the method step 50 within the scope of a third iteration (Iteration Nr. 3) for merging said adjacent data packages of step 230. Having merged said data package 1 and data package 2 to a new data package 1, the data records of the remaining new data package 1 are reorganized, and thereafter sorted again according to the given order for said granularity characteristics as illustrated in Table 1 and in Fig. 4b, respectively. The outcome of this reorganization and sorting is illustrated hereinafter in Table 7.

Table 7

Step: 250

rating method	rating-segment	Summe financial statement key figure 1	Minimum financial statement key figure 2	currency	old package	new package
Kreditinstitute	private banks	1140	166310	EUR	1	1
Kreditinstitute	public banks	6114	456825	EUR	1	1
Kreditinstitute	public banks	8149	484449	EUR	2	1
Versicherungen	life insurances	7586	670365	EUR	1	1
Versicherungen	life insurances	8715	247374	EUR	2	1
Versicherungen	casualty insurances	3346	729541	EUR	1	1
Versicherungen	casualty insurances	17599	35040	EUR	2	1

[0067] Thereupon, the aggregation for said key figures using said predetermined aggregation operations is conducted just once more by assigning said data records to the first parallel process illustrated by method step 40 of Fig. 2 within the scope of a third iteration (Iteration Nr. 3), wherein as a result, the size of the new data package 1 decreases from $M_m = 7$ to $N_a = 4$ according to the customer defined granularity as illustrated in Fig. 5 referring to "search result". Following the aggregation, the results of the remaining data packages are saved in a database. This outcome of this aggregation

is illustrated hereinafter in Table 8.

Table 8

Step: 250

rating method	rating-segment	financial statement key figure 1	financial statement key figure 2	currency	package
Kreditinstitute	private banks	1140	166310	EUR	1
Kreditinstitute	public banks	14263	456825	EUR	1
Versicherungen	casualty insurances	16301	247374	EUR	1
Versicherungen	casualty insurances	20945	35040	EUR	1

[0068] By checking the termination criterion for the loop i = i + 1 once again in step 260 ("not in parallel"), the aggregation algorithm 30 of Fig. 2 terminates at this point, since there is no other adjacent data package, whose first key is an element of any other data package, or in other words interpreting the termination criterion, all the data packages are disjoint with regard to the granularity characteristics.

[0069] Fig. 6 illustrates an example of use for an optimized aggregation algorithm compared to the preceding example of use of Fig. 5, using a processing tool based on the parallel aggregation algorithm 30 of Fig. 2. The aggregation of the raw data illustrated in Fig. 4a including M = 12 data records reduces the amount of data to 4 < M = 12 data records according to the customer defined granularity, as it is set out in Fig. 5 referring to "search result".

[0070] The granularity fields including granularity characteristics are characterized by "rating method" and "rating segment". The fields of key figures are characterized by the columns "financial statement key figure 1" and "financial statement key figure 2".

[0071] The data package size is determined through customizing. According to the statement that large data package sizes are ideal for aggregating, whereas small data package seizes are ideal for reorganizing, in this example of use the data package size (M_p) for aggregating is determined relatively great with $M_p = 8$ and the sub data package size (N_{sp}) is determined relatively low with $N_{sp} = 3$, thereby complying with the interests working in opposite direction.

[0072] In method step 200, the raw data shown in the original work list of Fig. 4a are exemplarily arranged by the key figures in the column "financial statement key figure 1" in ascending order so as to demonstrate a work list of unsorted records to begin with. Due to the customized data package size of $M_p = 8$, the M = 12 data records of said work list are split in two data packages, in data package 1 for aggregating, including $M_p = 8$ data records, and a remaining data package 2 corresponding to a remaining rest that is not to be aggregated, including 4 data records.

[0073] In analogy to the preceding example of use in Fig. 5, for the exemplification of the parallel aggregation algorithm as illustrated in Fig. 2 on the basis of the concrete example and to simplify matters, only the granularity fields characterized by "rating method" and "rating segment", the fields of key figures characterized by the columns "financial statement key figure 1" and "financial statement key figure 2", and the field currency are taken into consideration. All the rest of fields remain empty. Hereinafter, Table 9 illustrates the outcome of this reorganization and simplification of said original work list shown in Fig. 4a.

Table 9

Step: 200

rating method	rating-segment	Summe financial statement key figure 1	Minimum financial statement key figure 2	currency	package
insurances	life insurances	1620	865860	EUR	1
credit institutions	private banks	1912	809485	EUR	1
credit institutions	public banks	2860	456825	EUR	1
credit institutions	public banks	3254	693677	EUR	1
insurances	casualty insurances	3346	729541	EUR	2
credit institutions	private banks	3393	542616	EUR	2
insurances	life insurances	5966	670365	EUR	2
credit institutions	private banks	6135	166310	EUR	2
credit institutions	public banks	8149	484449	EUR	3
insurances	casualty insurances	8683	824001	EUR	3
insurances	life insurances	8715	247374	EUR	3
insurances	casualty insurances	8916	35040	EUR	3

[0074] In method step 210, the data packages are assigned to the parallel processing of the aggregation algorithm 30 of Fig. 2. Within the scope of a first iteration (Iteration Nr. 1), the parallel processing begins with the method step 40 of Fig. 2. Up to this point, there is not identified one granularity level i mentioned above, which is symbolized by i = 0 in Fig. 2. Therefore, at first, the fields of granularity characteristics labeled "rating method" and "rating segment" are identified so as to obtain levels i of granularity characteristics within said data packages, thereby accomplishing the criterion i > 0. The maximum reachable number of granularity levels i per data package is i = 4 due to Table 1 illustrated in the preceding example of use of Fig.5.

[0075] By sequentially comparing said customer defined granularity characteristics shown in Table 1 with the data records of each of the two data packages, thereby beginning with the first row of granularity characteristics of Table 1 characterized through "credit institutions / private banks" and ending with the forth row characterized through "insurances / casualty insurances", the data records of data package 1 and data package 2 are searched for matching results. In our example of use there appear in total four matches in data package 1 and three matches in data package 2 in respect to said granularity characteristics and granularity levels i, respectively, what as a result corresponds to a granularity level of i = 4 of identified granularity characteristics for data package 1 and i = 3 for data package 2.

[0076] Subsequently, both of the data packages are sorted according to the given order as set out in Table 1 of the preceding example of use of Fig. 5. The outcome of this sorting of data packages is illustrated below in Table 10.

Table 10

Step: 200

rating method	rating-segment	Summe financial statement key figure 1	Minimum financial statement key figure 2	currency	package
credit institutions	private banks	1912	809485	EUR	1
credit institutions	private banks	3393	542616	EUR	1
credit institutions	private banks	6135	166310	EUR	1
credit institutions	public banks	2860	456825	EUR	1
credit institutions	public banks	3254	693677	EUR	1
insurances	life insurances	1620	865860	EUR	1
insurances	life insurances	5966	670365	EUR	1
insurances	casualty insurances	3346	729541	EUR	1
credit institutions	public banks	8149	484449	EUR	2
insurances	life insurances	8715	247374	EUR	2
insurances	casualty insurances	8683	824001	EUR	2
insurances	casualty insurances	8916	35040	EUR	2

[0077] Subsequently, the rows of data package 1 are aggregated for the key figures (Xj) characterized through "financial statement key figure 1 " and "financial statement key figure 2" by applying appropriate aggregation operations (operators j) to the respective key figures, wherein said aggregation operations being predetermined or customized aggregation operations. In this case the matches are added up in respect to key figure 1, and in respect to key figure 2 the minimum value is taken over, thereby reducing the number of data records. As a result, data package 1 is reduced to $N_a = 4 < M_p = 8$ data records, which is illustrated below in Table 11.

Table 11

Step: 210

rating method	rating-segment	Summe financial statement key figure 1	Minimum financial statement key figure 2	currency	package
credit institutions	private banks	11440	166310	EUR	1
credit institutions	public banks	6114	456825	EUR	1
insurances	life insurances	7586	670365	EUR	1
insurances	casualty insurances	3346	729541	EUR	1
credit institutions	public banks	8149	484449	EUR	2
insurances	life insurances	8715	247374	EUR	2
insurances	casualty insurances	8683	824001	EUR	2
insurances	casualty insurances	8916	35040	EUR	2

[0078] Thereafter, the data packages are split into sub data packages and then the sub data packages are saved in a database. Since the sub data package size (N_{sp}) is determined by $N_{sp} = 3$, each of the two remaining data packages including 4 data records is split in two sub data packages, wherein each of the sub data packages 1 and 3 includes 3 data records, and each of the sub data packages 2 and 4 corresponding to the rest of data package 1 and data package 2, respectively, only includes 1 data record. The outcome of this splitting of data packages into sub data packages is

illustrated below in Table 12. Table 12

Table 12

Step: 210

rating method	rating-segment	financial statement key figure 1	financial statement key figure 2	currency	package old	package new
credit institutions	private banks	11440	166310	EUR	1	1
credit institutions	public banks	6114	456825	EUR	1	1
insurances	life insurances	7586	670365	EUR	1	1
insurances	casualty insurances	3346	729541	EUR	1	2
credit institutions	public banks	8149	484449	EUR	2	3
insurances	life insurances	8715	247374	EUR	2	3
insurances	casualty insurances	8683	824001	EUR	2	3
insurances	casualty insurances	8916	35040	EUR	2	4

[0079] Thereupon, in step 220, the identification of adjacent data packages based on these small sub data packages is conducted by checking the termination criterion of the loop i = i + 1 ("not in parallel") being:
if $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate,**
wherein *pos1* illustrates the first position of a data package, ***posmax*** illustrates the last position of a data package, and x, y illustrates the number of a data package, thereby comparing the key of the first record of each data package with the key of the first and the last record of all the rest of data packages (thus comparing all combinations x, y). If said criterion is not accomplished, meaning that the conditional inquiry is true, the data packages are assigned for rebuilding new data packages.

[0080] Beginning with the key of the first record of data package 1, the comparison of data package 1 and data package 2 results in that the key of the first record of data package 1 is less than the key of the single record of data package 2. As a result, interpreting the conditional inquiry for the loop i = i + 1, data package 1 and data package 2 do not intersect. Accordingly, data package 1 and data package 4 do not intersect. In contrast, data package 1 and data package 3 appear to intersect, since the key of the first record of data package 2 is greater than the key of the first record of data package 1 and less than the key of the last record of data package 1, meaning that the key of the first record of data package 2 is an element of the amount of data records in data package 1. Thus, they are identified as adjacent data packages. Further, the keys of data package 2 and 4 are identical, and thus both packages are not disjoint. Accordingly, they are identified as adjacent data packages, too. Thereupon, the data packages identified as adjacent are assigned for rebuilding new data packages.

[0081] Since data package 1 and data package 2, respectively, only include 3 records, the data package size of the new data package 1 including 6 records is still less than the determined package size of $M_p = 8$. The data package size of the new data package 2 includes 1 + 1 = 2 data records. In order to restore the original package size of $M_p = 8$, the new data package 2 is additionally added to the new data package 1.

[0082] In step 230, the sub data packages of step 210 are assigned to the second parallel processing of the aggregation algorithm 30 of Fig. 2 illustrated by the method step 50 within the scope of a second iteration (Iteration Nr. 2) for merging adjacent data packages and rebuilding new data packages, respectively. Thus, having merged sub data package 1 with sub data package 3, and sub data package 2 with sub data package 4, and additionally added the new data package 2 to the new data package 1, in all only one new data package remains. Subsequent to the merger, the data records are assigned again to the above mentioned first parallel process illustrated by method step 40 of Fig. 2 within the scope of a second iteration (Iteration Nr. 2), wherein the data records of the remaining new data package 1 are reorganized, and thereafter sorted again according to the given order of said granularity characteristics as illustrated in Table 1 of the preceding example of use and in Fig. 4b, respectively. The outcome of this reorganization and sorting is illustrated hereinafter in Table 13.

Table 13

Step: 230

rating method	rating-segment	Summe financial statement key figure 1	Minimum financial statement key figure 2	currency	package old	package new
credit institutions	private banks	11440	166310	EUR	1	1
credit institutions	public banks	6114	456825	EUR	1	1
credit institutions	public banks	8149	484449	EUR	3	1
insurances	life insurances	7586	670365	EUR	1	1
insurances	life insurances	8715	247374	EUR	3	1
insurances	casualty insurances	3346	729541	EUR	2	1
insurances	casualty insurances	8683	824001	EUR	3	1
insurances	casualty insurances	8916	35040	EUR	4	1

[0083] Thereupon, the aggregation for said key figures using said predetermined aggregation operations is conducted just once more, wherein as a result, the size of the new data package 1 decreases from $M_p = 8$ to $N_a = 4$ according to the customer defined granularity as illustrated in Fig. 5 referring to "search result". This outcome of this aggregation is illustrated hereinafter in Table 14.

Table 14

Step: 230

rating method	rating-segment	financial statement key figure 1	financial statement key figure 2	currency	package
credit institutions	private banks	11440	166310	EUR	1
credit institutions	public banks	14263	456825	EUR	1
insurances	life insurances	16301	247374	EUR	1
insurances	casualty insurances	20945	35040	EUR	1

[0084] Following the aggregation, the results of the remaining data package are saved in a database.

[0085] By checking the termination criterion for the loop i = i + 1 once again in step 240, the aggregation algorithm 30 of Fig. 2 terminates at this point, since there is no other adjacent data package, whose first key is an element of any other data package, or in other words interpreting the termination criterion, all the data packages are disjoint with regard to the granularity characteristics.

[0086] It will be apparent to those skilled in the art that various modifications and variations can be made in the system and method of the present invention without departing from the spirit and scope of the invention. Thus, it is intended that the present invention covers the modifications and variations of this invention provided that they come within the scope of the appended claims and their equivalents.

Claims

1. A computer-implemented method for automated generic and parallel aggregation of characteristics and key figures of mass data, said mass data including M records from a single database of a single data source or from different databases of different data sources, particularly associated with financial institutions and with financial affairs in banking practice, and further including sets of rows and sets of columns, each row corresponding to a record, and

the columns including fields of predetermined granularity characteristics and fields of predetermined key figures, wherein said aggregation reduces the amount of data to $N \leq M$ records for a customer defined granularity, the method comprising the following steps:

receiving said mass data from a single database of a single data source or from different databases of different data sources associated with banking practice;
selecting predetermined granularity characteristics and predetermined key figures, and selecting predetermined aggregation operations to be carried out by the processing means of a data processing system;
reading input data from a single database of a single data source or from different databases of different data sources into the processing means of a data processing system;
preparing the input data as data packages being of the size M_p in a preparational step before the aggregation starts;
processing the data packages being of the size M_p in a parallel process by identifying said granularity characteristics, thereby identifying unique granularity levels i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the records in each data package for key figures by using aggregation operations; and
following the aggregation, saving the results of each data package.

2. The method of claim 1, wherein the aggregation is computed for said predetermined granularity characteristics and / or predetermined key figures using predetermined aggregation operations selected from a function pool and / or costumer defined aggregation operations to be defined by input means using said predetermined aggregation operations.

3. The method of claim 1, wherein the aggregation is computed for costumer defined granularity characteristics and / or costumer defined key figures that are to be defined by input means using said predetermined aggregation operations selected from a function pool and using said predetermined aggregation operations and / or said costumer defined aggregation operations.

4. The method of claim 1, wherein said data packages being of the size M_p are processed within a loop i = i + 1 comprising the steps of:

a first parallel process for identifying said granularity characteristics, thereby identifying unique granularity levels i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the data records in each data package for key figures by using aggregation operations; thereby reducing the amount of data records to $N_a \leq M_p$; and following the aggregation, saving the results of each data package in a local database and storing the size and the key of the first and the last record of each data package in a global database; and subsequently checking the termination criterion for the loop i = i + 1 ("not in parallel") being:
if $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate,**

wherein *pos1* illustrates the first position of a data package, *posmax* illustrates the last position of a data package, and *x, y* illustrates the number of data package, and if the conditional criterion is not accomplished for all combinations (x, y), meaning that the conditional inquiry is true, thereby comparing the key of the first record of each data package with the first and the last record of all the rest of packages, assigning the aggregated packages to a second parallel process for merging adjacent data packages so as to rebuild new data packages, wherein adjacent packages are those packages with keys of the first record which are closest together and have violated the termination criterion, then storing the merged packages to a local database, and subsequently assigning the merged data packages again to the above mentioned first parallel process for reorganizing and sorting said new data packages, and thereafter aggregating said new data packages for key figures by using aggregation operations, and following the aggregation, after each loop cycle checking the termination criterion for the loop i = i + 1 for all combinations (x, y) anew, repeatedly executing the loop i = i + 1 while the termination criterion for the loop is not accomplished, else after accomplishing said criterion, i.e. all the data packages are disjoint with regard to the granularity characteristics, terminating the loop i = i + 1.

5. The method of claim 1, wherein said data packages being of the size M_p are processed within a loop i = i + 1 comprising the steps of:

a first parallel process for identifying said granularity characteristics, thereby identifying unique granularity levels

i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the data records in each data package for key figures by using aggregation operations; thereby reducing the amount of data records to $N_a \leq M_p$, and following the aggregation, splitting the aggregated data packages into several smaller data sub packages being of the size N_{sp} and saving the results of each sub data package in a local database; storing the size and the key of the first and the last record of each sub data package to a global database; and subsequently identifying adjacent packages based on these small sub data packages by checking ("not in parallel") the termination criterion for the loop i = i + 1 being: **if** $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate,**

wherein *pos1* illustrates the first position of a data package, *posmax* illustrates the last position of a data package, and *x, y* illustrates the number of data package, and if the conditional criterion is not accomplished for all combinations (x, y), meaning that the conditional inquiry is true, thereby comparing the key of the first record of each sub data package with the first and the last record of all the rest of sub data packages, assigning the sub data packages to a second parallel process for merging adjacent sub data packages so as to rebuild new data packages, wherein adjacent sub data packages are those data packages with keys of the first record which are closest together and have violated the termination criterion, and wherein by merging said sub data packages the original package size N is restored; then storing the new data packages to a local database; and subsequently assigning the new data packages again to the above mentioned first parallel process for reorganizing and sorting; and thereafter aggregating said new data packages for key figures by using aggregation operations; and following the aggregation, after each loop cycle splitting the aggregated data packages again into several smaller sub data packages and saving the results of each sub data package in a local database; storing the size and the key of the first and the last record of each sub data package to a global database; and subsequently identifying adjacent packages again based on these small packages by checking ("not in parallel") the termination criterion for the loop i = i + 1 for all combinations (x, y) anew, repeatedly executing the loop i = i + 1 while the termination criterion for the loop is not accomplished, else after accomplishing said criterion, i.e. all the data packages are disjoint with regard to the granularity characteristics, terminating the loop i = i + 1.

6. The method of claim 4 or 5, wherein ultimately conducting an additional calculation step for enriching aggregated data packages, and subsequently saving the data packages to a global result database.

7. The method of claim 1, further comprising the steps of:

enriching said prepared data packages in a parallel pre-processing step via secondary data source or data sources before the parallel aggregation starts; and
saving the results to a local database.

8. The method of claim 6, further comprising the steps of:

enriching the aggregated data packages in a parallel post-processing step via secondary data source or data sources following the parallel aggregation; and
saving the results to a global result database.

9. The method of claim 1, wherein the data packages are processed within jobs, each of the jobs including a plurality of data packages.

10. The method of claim 9, wherein one job or a plurality of jobs are processed in a parallel processing mode using a single data processor.

11. The method of claim 9, wherein one job or a plurality of jobs are processed in a parallel processing mode using a network of data processors.

12. The method of claim 11, wherein the network of data processors is a Local Area Network (LAN), Wide Area Network (WAN), intranet or internet.

13. The method of claims 1, wherein said data packages are processed within jobs, and wherein the jobs are processed in a parallel processing mode using a single data processor, thereby aggregating and merging the data packages of a job sequentially.

14. The method of claims 1, wherein said data packages are processed within jobs, and wherein the jobs are processed in a parallel processing mode using a network of data processors, thereby aggregating and merging the data packages of a job sequentially.

15. A computer system configured to perform automated generic and parallel aggregation of characteristics and key figures of mass data, said mass data including M records from a single database of a single data source or from different databases of different data sources, particularly associated with financial institutions and with financial affairs in banking practice, and further including sets of rows and sets of columns, each row corresponding to a record, and the columns including fields of predetermined granularity characteristics and fields of predetermined key figures, wherein said aggregation reduces the amount of data to $N \leq M$ records for a customer defined granularity, the method comprising the following steps:

receiving said mass data from a single database of a single data source or from different databases of different data sources associated with banking practice;
selecting predetermined granularity characteristics and predetermined key figures, and selecting predetermined aggregation operations to be carried out by the processing means of a data processing system;
reading input data from a single database of a single data source or from different databases of different data sources into the processing means of a data processing system;
preparing the input data as data packages being of the size M_p, in a preparational step before the aggregation starts;
processing the data packages being of the size M_p in a parallel process by identifying said granularity characteristics, thereby identifying unique granularity levels i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the records in each data package for key figures by using aggregation operations; and
following the aggregation, saving the results of each data package.

16. A computer system of claim 15, wherein said processing means are configured to process the data packages being of the size M_p within a loop i = i + 1 including the steps of:

a first parallel process for identifying said granularity characteristics, thereby identifying unique granularity levels i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the data records in each data package for key figures by using aggregation operations; thereby reducing the amount of data records to $N_a \leq M_p$; and following the aggregation, saving the results of each data package in a local database and storing the size and the key of the first and the last record of each data package in a global database; and subsequently checking the termination criterion for the loop i = i + 1 ("not in parallel") being:
if $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate,**

wherein *pos1* illustrates the first position of a data package, *posmax* illustrates the last position of a data package, and *x, y* illustrates the number of data package, and if the conditional criterion is not accomplished for all combinations (x, y), meaning that the conditional inquiry is true, thereby comparing the key of the first record of each data package with the first and the last record of all the rest of packages, assigning the aggregated packages to a second parallel process for merging adjacent data packages so as to rebuild new data packages, wherein adjacent packages are those packages with keys of the first record which are closest together and have violated the termination criterion, then storing the merged packages to a local database, and subsequently assigning the merged data packages again to the above mentioned first parallel process for reorganizing and sorting said new data packages, and thereafter aggregating said new data packages for key figures by using aggregation operations, and following the aggregation, after each loop cycle checking the termination criterion for the loop i = i + 1 for all combinations (x, y) anew, repeatedly executing the loop i = i + 1 while the termination criterion for the loop is not accomplished, else after accomplishing said criterion, i.e. all the data packages are disjoint with regard to the granularity characteristics, terminating the loop i = i + 1.

17. A computer system of claim 15, wherein said processing means are configured to process the data packages being of the size M_p within a loop i = i + 1 including the steps of:

a first parallel process for identifying said granularity characteristics, thereby identifying unique granularity levels i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the data records in each data package for key figures by using

aggregation operations; thereby reducing the amount of data records to $N_a \leq M_p$; and following the aggregation, splitting the aggregated packages into several smaller sub packages and saving the results of each sub data package in a local database; storing the size and the key of the first and the last record of each sub data package to a global database; and subsequently identifying adjacent packages based on these small sub data packages by checking ("not in parallel") the termination criterion for the loop i = i + 1 being:
if $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate,**

wherein *pos1* illustrates the first position of a data package, *posmax* illustrates the last position of a data package, and *x, y* illustrates the number of data package, and if the conditional criterion is not accomplished for all combinations (x, y), meaning that the conditional inquiry is true, thereby comparing the key of the first record of each sub data package with the first and the last record of all the rest of sub data packages, assigning the sub data packages to a second parallel process for merging adjacent sub data packages so as to rebuild new data packages, wherein adjacent sub data packages are those data packages with keys of the first record which are closest together and have violated the termination criterion, and wherein by merging said sub data packages the original package size N is restored; then storing the new data packages to a local database; and subsequently assigning the new data packages again to the above mentioned first parallel process for reorganizing and sorting; and thereafter aggregating said new data packages for key figures by using aggregation operations; and following the aggregation, after each loop cycle splitting the aggregated data packages again into several smaller sub data packages and saving the results of each sub data package in a local database; storing the size and the key of the first and the last record of each sub data package to a global database; and subsequently identifying adjacent packages again based on these small packages by checking ("not in parallel") the termination criterion for the loop i = i + 1 for all combinations (x, y) anew, repeatedly executing the loop i = i + 1 while the termination criterion for the loop is not accomplished, else after accomplishing said criterion, i.e. all the data packages are disjoint with regard to the granularity characteristics, terminating the loop i = i + 1.

18. A computer system of claim 16 or 17, wherein said processing means are further configured to ultimately conduct an calculation step for enriching data packages, and wherein said storage means are further configured to subsequently save the data packages to a global result database.

19. A computer system of claim 15, wherein said processing means are further configured to ultimately enrich said prepared data packages in a parallel pre-processing step via secondary data source or data sources before the parallel aggregation starts, and wherein said storage means are further configured to save the results to said local database.

20. A computer system of claim 18, wherein said processing means are further configured to ultimately enrich the aggregated data packages in a parallel post-processing step via secondary data source or data sources following the parallel aggregation, and wherein said storage means are further configured to save the results to said global database.

21. A computer program product having a plurality of instructions for causing processing means of a computer system to execute the following steps:

receiving said mass data from a single database of a single data source or from different databases of different data sources associated with banking practice;
selecting predetermined granularity characteristics and predetermined key figures, and selecting predetermined aggregation operations to be carried out by the processing means of a data processing system;
reading input data from a single database of a single data source or from different databases of different data sources into the processing means of a data processing system;
preparing the input data as data packages being of the size M_p in a preparational step before the aggregation starts;
processing the data packages being of the size M_p in a parallel process by identifying said granularity characteristics, thereby identifying unique granularity levels i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the records in each data package for key figures by using aggregation operations; and
following the aggregation, saving the results of each data package.

22. The computer program product of claim 21, wherein the program comprises instructions for processing the data packages being of the size M_p within a loop i = i + 1 including the steps of:

a first parallel process for identifying said granularity characteristics, thereby identifying unique granularity levels i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the data records in each data package for key figures by using aggregation operations; thereby reducing the amount of data records to $N_a \leq M_p$; and following the aggregation, saving the results of each data package in a local database and storing the size and the key of the first and the last record of each data package in a global database; and subsequently checking the termination criterion for the loop i = i + 1 ("not in parallel") being:
if $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate,**

wherein *pos1* illustrates the first position of a data package, *posmax* illustrates the last position of a data package, and *x, y* illustrates the number of data package, and if the conditional criterion is not accomplished for all combinations (x, y), meaning that the conditional inquiry is true, thereby comparing the key of the first record of each data package with the first and the last record of all the rest of packages, assigning the aggregated packages to a second parallel process for merging adjacent data packages so as to rebuild new data packages, wherein adjacent packages are those packages with keys of the first record which are closest together and have violated the termination criterion, then storing the merged packages to a local database, and subsequently assigning the merged data packages again to the above mentioned first parallel process for reorganizing and sorting said new data packages, and thereafter aggregating said new data packages for key figures by using aggregation operations, and following the aggregation, after each loop cycle checking the termination criterion for the loop i = i + 1 for all combinations (x, y) anew, repeatedly executing the loop i = i + 1 while the termination criterion for the loop is not accomplished, else after accomplishing said criterion, i.e. all the data packages are disjoint with regard to the granularity characteristics, terminating the loop i = i + 1.

23. The computer program product of claim 21, wherein the program comprises instructions for processing the data packages being of the size M_p within a loop i = i + 1 including the steps of:

a first parallel process for identifying said granularity characteristics, thereby identifying unique granularity levels i; sorting the records of each data package for a given order of granularity characteristics of said customized granularity; and subsequently aggregating the data records in each data package for key figures by using aggregation operations; thereby reducing the amount of data records to $N_a \leq M_p$, and following the aggregation, splitting the aggregated data packages into several smaller data sub packages being of the size N_{sp} and saving the results of each sub data package in a local database; storing the size and the key of the first and the last record of each sub data package to a global database; and subsequently identifying adjacent packages based on these small sub data packages by checking ("not in parallel") the termination criterion for the loop i = i + 1 being:
if $key_{pos1,x} \in (key_{pos1,y}; key_{posmax:y})$ **then continue else terminate,**

wherein *pos1* illustrates the first position of a data package, *posmax* illustrates the last position of a data package, and *x, y* illustrates the number of data package, and if the conditional criterion is not accomplished for all combinations (x, y), meaning that the conditional inquiry is true, thereby comparing the key of the first record of each sub data package with the first and the last record of all the rest of sub data packages, assigning the sub data packages to a second parallel process for merging adjacent sub data packages so as to rebuild new data packages, wherein adjacent sub data packages are those data packages with keys of the first record which are closest together and have violated the termination criterion, and wherein by merging said sub data packages the original package size N is restored; then storing the new data packages to a local database; and subsequently assigning the new data packages again to the above mentioned first parallel process for reorganizing and sorting; and thereafter aggregating said new data packages for key figures by using aggregation operations; and following the aggregation, after each loop cycle splitting the aggregated data packages again into several smaller sub data packages and saving the results of each sub data package in a local database; storing the size and the key of the first and the last record of each sub data package to a global database; and subsequently identifying adjacent packages again based on these small packages by checking ("not in parallel") the termination criterion for the loop i = i + 1 for all combinations (x, y) anew, repeatedly executing the loop i = i + 1 while the termination criterion for the loop is not accomplished, else after accomplishing said criterion, i.e. all the data packages are disjoint with regard to the granularity characteristics, terminating the loop i = i + 1.

24. The computer program product of claim 22 or 23, wherein the program ultimately conducts an additional calculation step for enriching aggregated data packages, and wherein the data packages are subsequently saved to a global result database.

25. The computer program product of claim 21, further comprising the steps of:

enriching said prepared data packages in a parallel pre-processing step via secondary data source or data sources before the parallel aggregation starts; and
saving the results to a local database.

26. The computer program product of claim 24, further comprising the steps of:

enriching the aggregated data packages in a parallel post-processing step via secondary data source or data sources following the parallel aggregation; and
saving the results to a global result database.

27. A data carrier readable by a computer, the data carrier storing a plurality of instructions implemented by computer program for causing the processing means of a computer system to execute the method of claim 1.

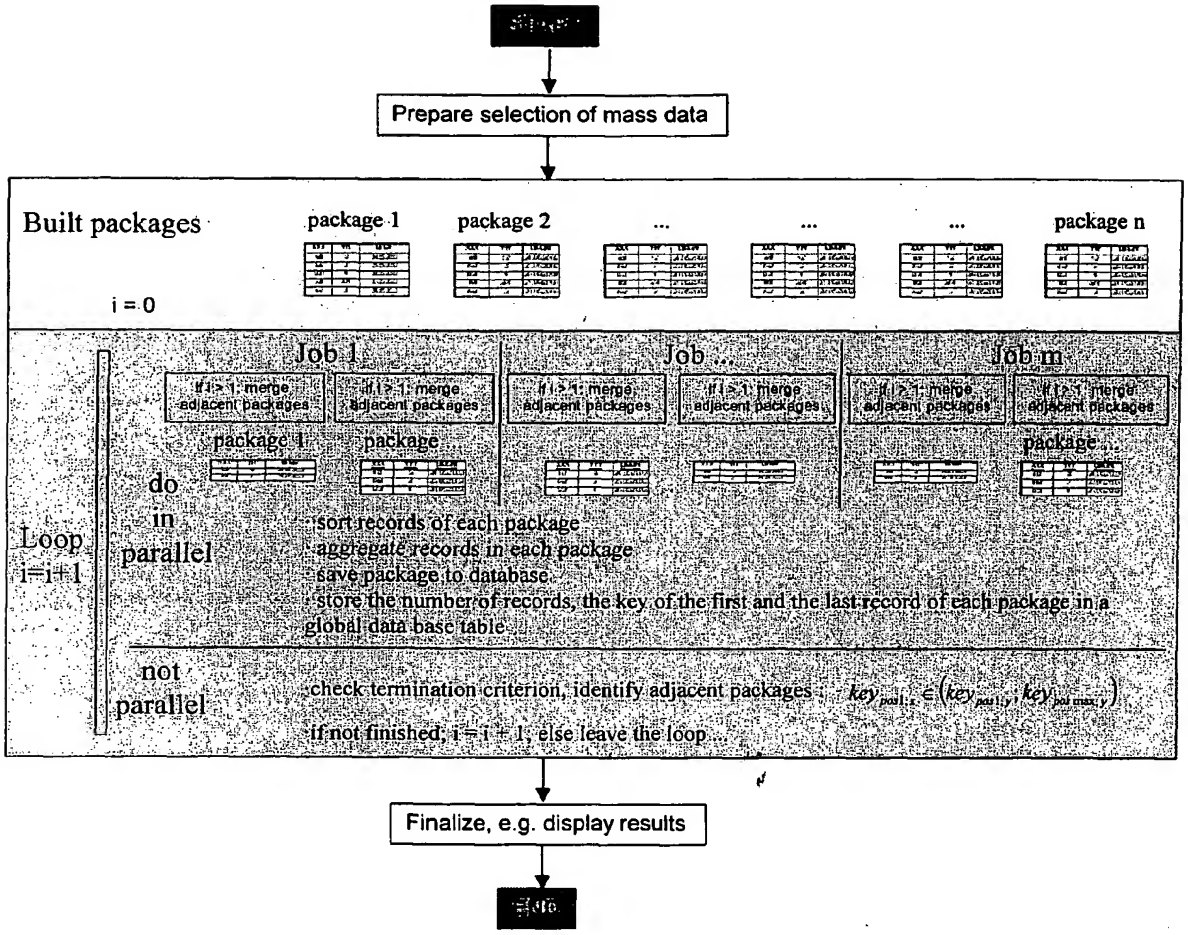
Prepare selection of mass data
Built packages
package 1
package 2
...
...
...
package n
i = 0
Job 1
Job ...
Job m
if i > 1: merge adjacent packages
package 1
package
package ...
Loop
i=i+1
do in parallel
sort records of each package
aggregate records in each package
save package to database
store the number of records, the key of the first and the last record of each package in a global data base table
not parallel
check termination criterion, identify adjacent packages
if not finished, i = i + 1, else leave the loop
Finalize, e.g. display results

Fig.1

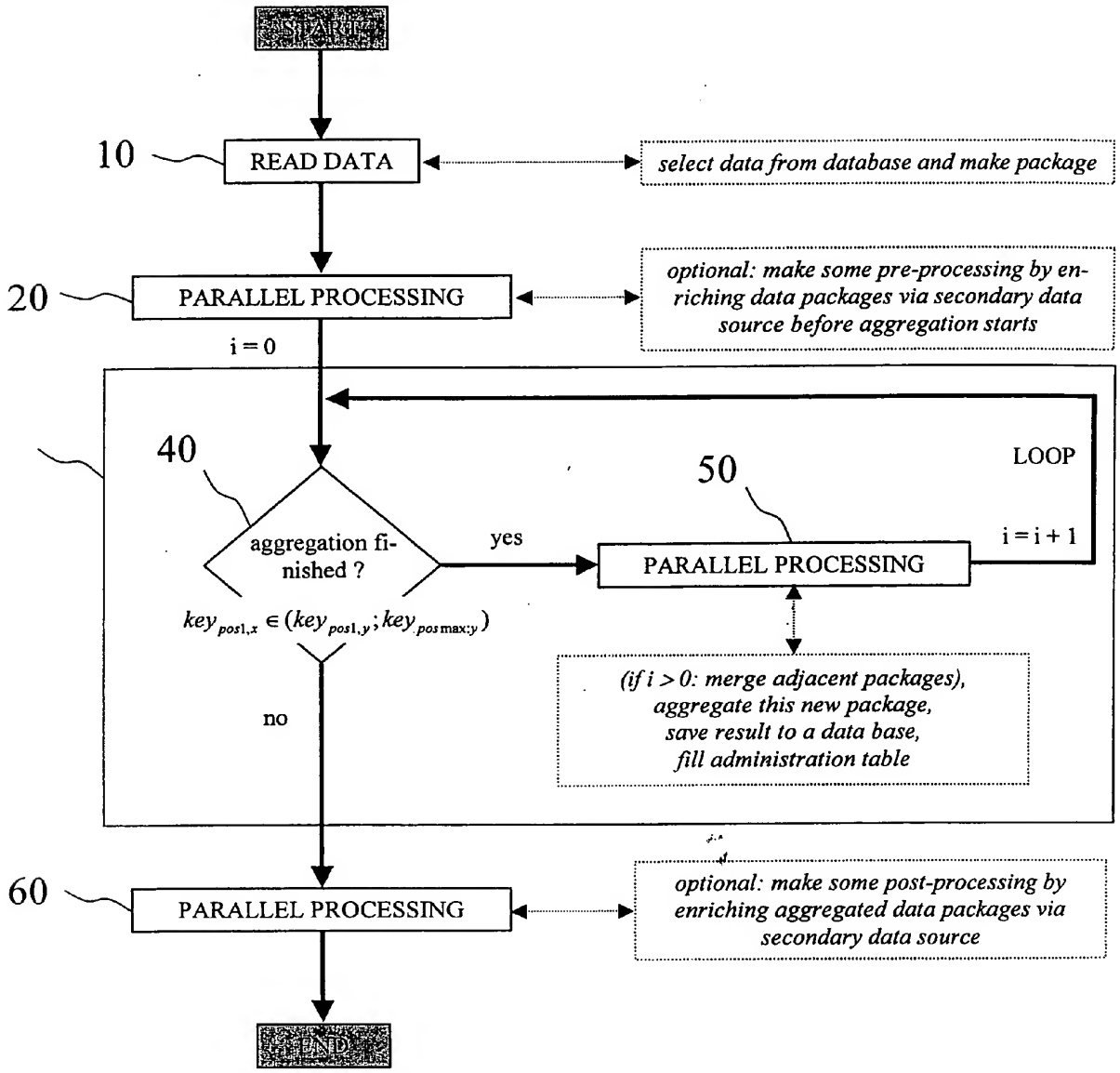
START
10
READ DATA
select data from database and make package
20
PARALLEL PROCESSING
optional: make some pre-processing by enriching data packages via secondary data source before aggregation starts
i = 0
40
aggregation finished ?
yes
50
PARALLEL PROCESSING
LOOP
i = i + 1
key pos1,x ∈ (key pos1,y ; key posmax;y)
(if i > 0: merge adjacent packages), aggregate this new package, save result to a data base, fill administration table
no
60
PARALLEL PROCESSING
optional: make some post-processing by enriching aggregated data packages via secondary data source
END

Fig. 2

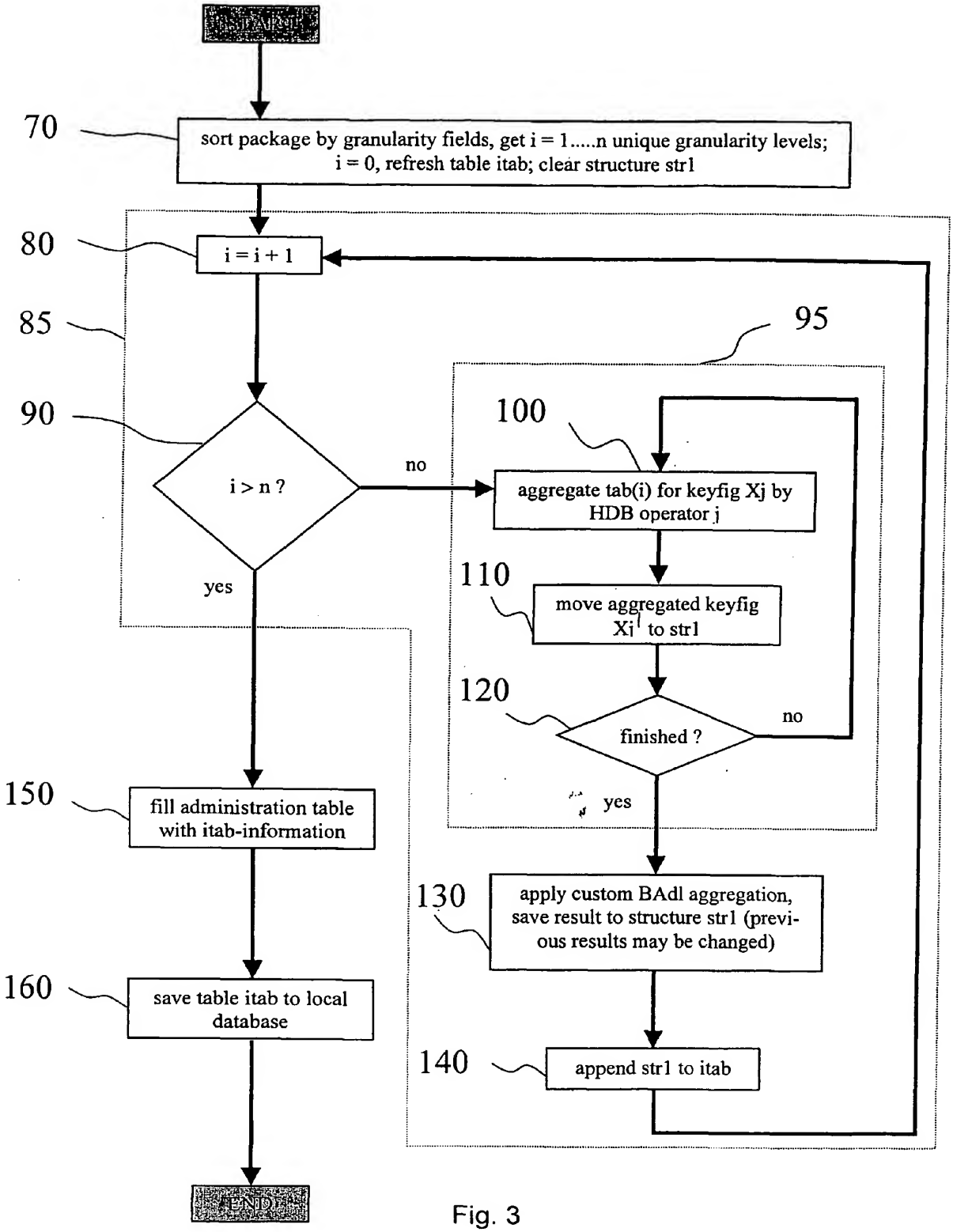
START
70
sort package by granularity fields, get i = 1.....n unique granularity levels;
i = 0, refresh table itab; clear structure str1
80
i = i + 1
85
95
90
i > n ?
no
yes
100
aggregate tab(i) for keyfig Xj by HDB operator j
110
move aggregated keyfig Xi' to str1
120
finished ?
no
yes
150
fill administration table with itab-information
130
apply custom BAdI aggregation, save result to structure str1 (previous results may be changed)
160
save table itab to local database
140
append str1 to itab
END

Fig. 3

Fig. 4a

fields of granularity characteristics;				fields of key figures;			
rating method	rating segment	business partner	rating date	rating classification	financial statement key figure 1	financial statement key figure 2	currency
credit institutions	public banks	GP_SK_01	1-Jan-2002	AAA	8149	484449	EUR
credit institutions	public banks	GP_SK_02	1-Jan-2002	BBB	2860	456825	EUR
credit institutions	public banks	GP_SK_03	1-Jan-2002	CCC	3254	693677	EUR
credit institutions	private banks	GP_LB_04	1-Jan-2002	AAA	1912	809485	EUR
credit institutions	private banks	GP_LB_05	1-Jan-2002	AAA	6135	166310	EUR
credit institutions	private banks	GP_LB_06	1-Jan-2002	BBB	3393	542616	EUR
insurances	life insurances	GP_LV_07	1-Jan-2002	BBB	8715	247374	EUR
insurances	life insurances	GP_LV_08	1-Jan-2002	BBB	5966	670365	EUR
insurances	life insurances	GP_LV_09	1-Jan-2002	AAA	1620	865860	EUR
insurances	casualty insurances	GP_UV_10	1-Jan-2002	BBB	8683	824001	EUR
insurances	casualty insurances	GP_UV_11	1-Jan-2002	BBB	3346	729541	EUR
insurances	casualty insurances	GP_UV_12	1-Jan-2002	CCC	8916	35040	EUR

Fig. 4b

granularity characteristics /
granularity levels i of granularity characteristics; i = 4;

credit institutions	private banks
credit institutions	public banks
insurances	life insurances
insurances	casualty insurances

EP 1 643 364 A1

Aggregation example:

package size 4;

granularity charactereistic
rating method
rating segment

keyfigure Xj	Operator j
financial statement key figure 1	Sum
financial statement key figure 2	Minimum

form of sorting	direction of sorting

search result

fields of granularity characteristics;					**fields of key figures;**		
rating method	**rating segment**	business partner	rating date	rating classification	**financial statement key figure 1**	**financial statement key figure 2**	**currency**
credit institutions	private banks				???	???	???
credit institutions	public banks				???	???	???
insurances	life insurances				???	???	???
insurances	casualty insurances				???	???	???

Fig. 5

EP 1 643 364 A1

Step: 200 raw data shown in the work list of Fig. 4 are exemplarily arranged by the key figures in the column "financial statement key figure 1", in ascending order so as to demonstrate a work list of unsorted records to begin with;

package size = 4;

rating method	rating-segment	business partner	rating date	rating classification	Sum financial statement key figure 1	Minimum financial statement key figure 2	currency	package
insurances	life insurances	GP_LV_09	1-Jan-2002	AAA	1620	865860	EUR	1
credit institutions	private banks	GP_LB_04	1-Jan-2002	AAA	1912	809485	EUR	1
credit institutions	public banks	GP_SK_02	1-Jan-2002	BBB	2860	456825	EUR	1
credit institutions	public banks	GP_SK_03	1-Jan-2002	CCC	3254	693677	EUR	1
insurances	casualty insurances	GP_UV_11	1-Jan-2002	BBB	3346	729541	EUR	2
credit institutions	private banks	GP_LB_06	1-Jan-2002	BBB	3393	542616	EUR	2
insurances	life insurances	GP_LV_08	1-Jan-2002	BBB	5966	670365	EUR	2
credit institutions	private banks	GP_LB_05	1-Jan-2002	AAA	6135	166310	EUR	2
credit institutions	public banks	GP_SK_01	1-Jan-2002	AAA	8149	484449	EUR	3
insurances	casualty insurances	GP_UV_10	1-Jan-2002	BBB	8683	824001	EUR	3
insurances	life insurances	GP_LV_07	1-Jan-2002	BBB	8715	247374	EUR	3
insurances	casualty insurances	GP_UV_12	1-Jan-2002	CCC	8916	35040	EUR	3

Step: 210 Iteration Nr. 1 sorting within packages according to customized granularity levels, subsequently aggregating;

rating method	rating-segment	business partner	rating date	rating classification	Sum financial statement key figure 1	Minimum financial statement key figure 2	currency	package
credit institutions	private banks				1912	809485	EUR	1
credit institutions	public banks				6114	456825	EUR	1
insurances	life insurances				1620	865860	EUR	1
credit institutions	private banks				9528	166310	EUR	2
insurances	life insurances				5966	670365	EUR	2
insurances	casualty insurances				3346	729541	EUR	2
credit institutions	public banks				8149	484449	EUR	3
insurances	life insurances				8715	247374	EUR	3
insurances	casualty insurances				17599	35040	EUR	3

Step: 220 not in parallel comparison of packages and assignment for rebuilding;

a key of the last record of data package 1 is greater than the key of the first record of data package 2, and key of the first record of data package 1 is less than the key of the first record of data package 2, or in other words interpreted, the key of the first record of data package 2 is an element of data package 1;
package size: data package 1 includes 3 records, data package 2 includes 3 records, new data package 1 includes 6 records (exceedance of max. package size is acceptable)

b data package 3 remains;

Step: 230 Iteration Nr. 2 parallel reorganization of records, thereafter sorting of records within packages;

rating method	rating-segment	business partner	rating date	rating classification	Summe financial statement key figure 1	Minimum financial statement key figure 2	currency	old package	new package
credit institutions	private banks				1912	809485	EUR	1	1
credit institutions	private banks				9528	166310	EUR	2	1
credit institutions	public banks				6114	456825	EUR	1	1
insurances	life insurances				1620	865860	EUR	1	1
insurances	life insurances				5966	670365	EUR	2	1
insurances	casualty insurances				3346	729541	EUR	2	1
credit institutions	public banks				8149	484449	EUR	3	2
insurances	life insurances				8715	247374	EUR	3	2
insurances	casualty insurances				17599	35040	EUR	3	2

Iteration Nr. 2 now aggregating;

rating method	rating-segment	business partner	rating date	rating classification	financial statement key figure 1	financial statement key figure 2	currency	package
credit institutions	private banks				11400	166310	EUR	1
credit institutions	public banks				6114	456825	EUR	1
insurances	life insurances				7586	670365	EUR	1
insurances	casualty insurances				3346	729541	EUR	1
credit institutions	public banks				8149	484449	EUR	2
insurances	life insurances				8715	247374	EUR	2
insurances	casualty insurances				17599	35040	EUR	2

Step: 240 not in parallel comparison of packages and assignment for rebuilding;

key of the last record of data package 1 is greater than the key of the first record of data package 2, and key of the first record of data package 1 is less than the key of the first record of data package 2, or in other words interpreted, the key of the first record of data package 2 is an element of data package 1;
package size: data package 1 includes 4 records, data package 2 includes 3 records, new data package 1 includes 7 records (exceedance of max. package size is acceptable)

Step: 250 Iteration Nr. 3 parallel reorganization of records, thereafter sorting of records within packages;

rating method	rating-segment	business partner	rating date	rating classification	Summe financial statement key figure 1	Minimum financial statement key figure 2	currency	old package	new package
Kreditinstitute	private banks				1140	166310	EUR	1	1
Kreditinstitute	public banks				6114	456825	EUR	1	1
Kreditinstitute	public banks				8149	484449	EUR	2	1
Versicherungen	life insurances				7586	670365	EUR	1	1
Versicherungen	life insurances				8715	247374	EUR	2	1
Versicherungen	casualty insurances				3346	729541	EUR	1	1
Versicherungen	casualty insurances				17599	35040	EUR	2	1

Iteration Nr. 3 now aggregating;

rating method	rating-segment	business partner	rating date	rating classification	financial statement key figure 1	financial statement key figure 2	currency	package
Kreditinstitute	private banks				1140	166310	EUR	1
Kreditinstitute	public banks				14263	456825	EUR	1
Versicherungen	casualty insurances				16301	247374	EUR	1
Versicherungen	casualty insurances				20945	35040	EUR	1

Step: 260 not in parallel comparison of packages and assignment for rebuilding;

the aggregation algorithm terminates at this point, since there is no other adjacent data package remaining, whose first key is an element of any other data package;
in this case only one data package remains;
generally interpreting the termination criterion, all the data packages are disjoint with regard to the granularity characteristics;

Aggregation example:

package size for main packages 8; package size for sub packages 3;

granularity charactereistic
rating method
rating segment

keyfigure Xj	Operator j
financial statement key figure 1	Sum
financial statement key figure 2	Minimum

form of sorting	direction of sorting

search result

fields of granularity characteristics;					**fields of key figures;**		
rating method	**rating segment**	business partner	rating date	rating classification	**financial statement key figure 1**	**financial statement key figure 2**	**currency**
credit institutions	private banks				???	???	???
credit institutions	public banks				???	???	???
insurances	life insurances				???	???	???
insurances	casualty insurances				???	???	???

Fig. 6

EP 1 643 364 A1

Step: 200 raw data shown in the work list of Fig. 4 are exemplarily arranged by the key figures in the column "financial statement key figure 1" in ascending order so as to demonstrate a work list of unsorted records to begin with;
package size = 8, sub package size 3

rating method	rating-segment	business partner	rating date	rating classification	Summe financial statement key figure 1	Minimum financial statement currency key figure 2		package
insurances	life insurances	GP_LV_09	1-Jan-2002	AAA	1620	865860	EUR	1
credit institutions	private banks	GP_LB_04	1-Jan-2002	AAA	1912	809485	EUR	1
credit institutions	public banks	GP_SK_02	1-Jan-2002	BBB	2860	456825	EUR	1
credit institutions	public banks	GP_SK_03	1-Jan-2002	CCC	3254	693677	EUR	1
insurances	casualty insurances	GP_UV_11	1-Jan-2002	BBB	3346	729541	EUR	2
credit institutions	private banks	GP_LB_06	1-Jan-2002	BBB	3393	542616	EUR	2
insurances	life insurances	GP_LV_08	1-Jan-2002	BBB	5966	670365	EUR	2
credit institutions	private banks	GP_LB_05	1-Jan-2002	AAA	6135	166310	EUR	2
credit institutions	public banks	GP_SK_01	1-Jan-2002	AAA	8149	484449	EUR	3
insurances	casualty insurances	GP_UV_10	1-Jan-2002	BBB	8683	824001	EUR	3
insurances	life insurances	GP_LV_07	1-Jan-2002	BBB	8715	247374	EUR	3
insurances	casualty insurances	GP_UV_12	1-Jan-2002	CCC	8916	35040	EUR	3

Step: 210 Iteration Nr. 1 Sorting within packages for granularity characteristics, thereafter aggregating

rating method	rating-segment	business partner	rating date	rating classification	Summe financial statement key figure 1	Minimum financial statement currency key figure 2	package
credit institutions	private banks				11440	166310 EUR	1
credit institutions	public banks				6114	456825 EUR	1
insurances	life insurances				7586	670365 EUR	1
insurances	casualty insurances				3346	729541 EUR	1
credit institutions	public banks				8149	484449 EUR	2
insurances	life insurances				8715	247374 EUR	2
insurances	casualty insurances				8683	824001 EUR	2
insurances	casualty insurances				8916	35040 EUR	2

Iteration Nr. 1 Storing with sub package size 3

rating method	rating-segment	business partner	rating date	rating classification	financial statement key figure 1	financial statement currency key figure 2	package old	package new	
credit institutions	private banks				11440	166310 EUR	1	1	
credit institutions	public banks				6114	456825 EUR	1	1	
insurances	life insurances				7586	670365 EUR	1	1	
insurances	casualty insurances				3346	729541 EUR	1	2	rest of package 1
credit institutions	public banks				8149	484449 EUR	2	3	
insurances	life insurances				8715	247374 EUR	2	3	
insurances	casualty insurances				8683	824001 EUR	2	3	
insurances	casualty insurances				8916	35040 EUR	2	4	rest of package 2

Step: 220 not in parallel - comparison of packages and assignment for rebuilding;

a checking of disjoit packages;
data package 1 and data package 2 have no intersection;
data package 1 and data package 3 intersect, thus they are marked; new package size 3+3=6 (thus still < 8);
data package 1 and data package 4 have no intersection;
key of data package 2 is equal to key of data package 4, i.e. both packages are not disjoint;
thus data packages 2 and 4 are marked (new package size: 1 + 1 = 2 data records);

b restoring maximum package size;
the new package 1 + 3 includes only 6 data records, thus the new package 2 + 4 is added: in total 8 data records;
thus in all only 1 new data package is generated;

Step: 230 Iteration Nr. 2 parallel rebuilding of data packages, thereafter sorting within packages

rating method	rating-segment	business partner	rating date	rating classification	Summe financial statement key figure 1	Minimum financial statement currency key figure 2		package old	package new
credit institutions	private banks				11440	166310	EUR	1	1
credit institutions	public banks				6114	456825	EUR	1	1
credit institutions	public banks				8149	484449	EUR	3	1
insurances	life insurances				7586	670365	EUR	1	1
insurances	life insurances				8715	247374	EUR	3	1
insurances	casualty insurances				3346	729541	EUR	2	1
insurances	casualty insurances				8683	824001	EUR	3	1
insurances	casualty insurances				8916	35040	EUR	4	1

Iteration Nr. 2 now aggregating

rating method	rating-segment	business partner	rating date	rating classification	financial statement key figure 1	financial statement currency key figure 2		package
credit institutions	private banks				11440	166310	EUR	1
credit institutions	public banks				14263	456825	EUR	1
insurances	life insurances				16301	247374	EUR	1
insurances	casualty insurances				20945	35040	EUR	1

Step: 240 not in parallel comparison of packages and assignment for rebuilding,

the aggregation algorithm terminates at this point, since there is no other adjacent data package remaining, whose first key is an element of any other data package;
in this case only one data package remains;
generally interpreting the termination criterion, all the data packages are disjoint with regard to the granularity characteristics;

EP 1 643 364 A1

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 05 01 9072

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	LEINBERGER W ET AL: "Multi-Capacity Bin Packing Algorithms wit Applications to Job Scheduling under Multiple Constraints (TR 99-024)" UNIVERSITY OF MINNESOTA - COMPUTER SCIENCE AND ENGINEERING - TECHNICAL REPORT ABSTRACT, no. 99-24, 27 May 1999 (1999-05-27), XP002285342 * abstract * * paragraphs [0001] - [0003] * * paragraphs [04.3] - [04.5] *	1-27	G06F9/46
X	FIAT A ET AL: "Online Algorithms: the State of the Art passage" LECTURE NOTES IN COMPUTER SCIENCE, SPRINGER VERLAG, NEW YORK, NY, US, vol. 1442, 1998, pages 147-195, XP002294222 ISSN: 0302-9743 * page 147, line 1 - page 148, line 31 * * page 155, line 21 - page 160, line 39 * * page 161, line 26 - page 162, last line * * page 165, line 6 - page 166, line 34 * * page 178, line 1 - page 181, line 24 *	1-27	TECHNICAL FIELDS SEARCHED (Int.Cl.7) G06F
X	US 6 742 015 B1 (BOWMAN-AMUAH MICHEL K) 25 May 2004 (2004-05-25) * column 188, line 54 - column 200, line 2 * * column 302, line 19 - column 304, line 22 *	1-27	

-/--

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
The Hague	31 October 2005	Peelen, B

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 05 01 9072

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	MARUYAMA K ET AL: "A GENERAL PACKING ALGORITHM FOR MULTIDIMENSIONAL RESOURCE REQUIREMENTS" INTERNATIONAL JOURNAL OF INFORMATION AND COMPUTER SCIENCES, PLENUM PUBLISHING CORPORATION, US, vol. 6, no. 2, June 1977 (1977-06), pages 131-149, XP008032005 ISSN: 0091-7036 * abstract * * pages 131-137 *	1-27	
X	COFFMAN E G ET AL: "APPROXIMATION ALGORITHMS FOR BIN PACKINGA SURVEY" APPROXIMATION ALGORITHMS FOR NP-HARD PROBLEMS, 1996, pages 1-53, XP002285343 * abstract * * paragraphs [02.1] - [2.2.6] *	1-27	
A	GALAMBOS G ET AL: "REPACKING HELPS IN BOUNDED SPACE ON-LINE BIN-PACKING BEIM BOUNDED SPACE ON-LINE BIN-PACKING KANN UMPACKEN VON VORTEIL SEIN" COMPUTING, SPRINGER VERLAG, VIENNA, AT, vol. 49, no. 4, 1993, pages 329-338, XP008031985 ISSN: 0344-8029 * the whole document *	1-27	TECHNICAL FIELDS SEARCHED (Int.Cl.7)

-/--

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
The Hague	31 October 2005	Peelen, B

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 05 01 9072

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
A	AGGARWAL G ET AL: "The load rebalancing problem" SPAA 2003. 15TH. ANNUAL ACM SYMPOSIUM ON PARALLEL ALGORITHMS AND ARCHITECTURES. SAN DIEGO, CA, JUNE 7 - 9, 2003, ANNUAL ACM SYMPOSIUM ON PARALLEL ALGORITHMS AND ARCHITECTURES, NEW YORK, NY : ACM, US, 7 June 2003 (2003-06-07), pages 258-265, XP002285345 ISBN: 1-58113-661-7 * abstract * * paragraphs [0001] - [003.] * -----	1-27	

TECHNICAL FIELDS SEARCHED (Int.Cl.7)

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
The Hague	31 October 2005	Peelen, B

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 05 01 9072

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

31-10-2005

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 6742015 B1	25-05-2004	NONE	

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82